

# A Hybrid Method with Deviational Particles for Spatial Inhomogeneous Plasma\*

Bokai Yan<sup>†</sup>

December 10, 2015

## Abstract

In this work we propose a Hybrid method with Deviational Particles (HDP) for a plasma modeled by the inhomogeneous Vlasov-Poisson-Landau system. We split the distribution into a Maxwellian part evolved by a grid based fluid solver and a deviation part simulated by numerical particles. These particles, named deviational particles, could be both positive and negative. We combine the Monte Carlo method proposed in [31], a Particle in Cell method and a Macro-Micro decomposition method [3] to design an efficient hybrid method. Furthermore, coarse particles are employed to accelerate the simulation. A particle resampling technique on both deviational particles and coarse particles is also investigated and improved. This method is applicable all all regimes and significantly more efficient compared to a PIC-DSMC method near the fluid regime.

## 1 Introduction

The evolution of a class of collisional plasmas can be modeled by the spatially inhomogeneous Vlasov-Poisson-Landau (VPL) system, with the electromagnetic fields and applied magnetic fields absent,

$$\partial_t f + \mathbf{v} \cdot \nabla_{\mathbf{x}} f - \mathbf{E} \cdot \nabla_{\mathbf{v}} f = Q_L(f, f), \quad (1a)$$

$$-\nabla_{\mathbf{x}} \cdot \mathbf{E} = \rho(t, \mathbf{x}) = \int f(t, \mathbf{x}, \mathbf{v}) d\mathbf{v}. \quad (1b)$$

Here  $f(t, \mathbf{x}, \mathbf{v})$  is the time dependent density distribution function of the charged particles (electrons and ions) in the phase space  $(\mathbf{x}, \mathbf{v})$ , with  $\mathbf{x} \in \mathbb{R}^3$  the spatial coordinates and  $\mathbf{v} \in \mathbb{R}^3$  the velocity coordinates. The left hand side of (1a) describes the advection of particles under the electric field  $\mathbf{E}$ , which is solved by the Poisson equation (1b). On the right hand side, the Landau (or Landau-Fokker-Planck) operator,

$$Q_L(f, f) = \frac{A}{4} \frac{\partial}{\partial v_i} \int_{\mathbb{R}^3} u \sigma_{tr}(u) (u^2 \delta_{ij} - u_i u_j) \left( \frac{\partial}{\partial v_j} - \frac{\partial}{\partial w_j} \right) f(\mathbf{w}) f(\mathbf{v}) d\mathbf{w} \quad (2)$$

models the binary collisions due to the long range Coulomb interaction. Here  $u = |\mathbf{v} - \mathbf{w}|$ ,  $A = 2\pi \left( \frac{e^2}{2\pi\epsilon_0 m} \right)^2 \log \Lambda$ , with  $e$  the charge of an individual particle,  $m$  its mass,  $\epsilon_0$  the permittivity of free space, and  $\log \Lambda$  the Coulomb logarithm.

The small angle collisions dominate in Coulomb interaction. The Landau operator (2) is derived as the grazing limit of Boltzmann operator (see [10] for example) by Landau in 1936 [20]. We refer

---

\*This research was supported by DOE DE-FG02-13ER26152/DE-SC0010613.

<sup>†</sup>Mathematics Department, University of California at Los Angeles, Los Angeles, CA 90095-1555 USA.  
byan@math.ucla.edu.

the reader to Villani [29] and the references therein for a review of the mathematical derivation. The Landau operator inherits many properties of the Boltzmann operator. First, the density, momentum and energy are conserved during the collision process, i.e.

$$\langle \phi Q \rangle = 0, \quad \text{for } \phi(\mathbf{v}) = 1, \mathbf{v}, |\mathbf{v}|^2/2,$$

where  $\langle \cdot \rangle$  is defined as  $\langle \psi \rangle = \int_{\mathbb{R}^3} \psi(\mathbf{v}) d\mathbf{v}$ .

Furthermore, the entropy is dissipated

$$\langle Q \log f \rangle \leq 0,$$

with the equal sign holds if and only if  $f$  is the local Maxwellian

$$M(t, \mathbf{x}, \mathbf{v}) = \frac{\rho(t, \mathbf{x})}{(2\pi T(t, \mathbf{x}))^{3/2}} e^{-\frac{|\mathbf{v} - \mathbf{u}(t, \mathbf{x})|^2}{2T(t, \mathbf{x})}}. \quad (3)$$

Here the density  $\rho$ , macroscopic velocity  $\mathbf{u}$  and temperature  $T$  are determined by  $f$  via

$$\langle \phi M(t, \mathbf{x}, \mathbf{v}) \rangle = \langle \phi f(t, \mathbf{x}, \mathbf{v}) \rangle, \quad \text{for every } \mathbf{x}, t, \quad (4)$$

with  $\phi = 1, \mathbf{v}, |\mathbf{v}|^2/2$ .

The numerical schemes for the VPL system (1) have been studied extensively. Depending on how to represent the velocity space, these methods can be categorized into two classes, the deterministic (also known as grids based or continuum) methods and probabilistic (also known as particles based) methods. Various techniques have been developed for deterministic methods. We refer the user to [15] and the references therein for a recent review. However all the deterministic methods are suffered from the *curse of dimensionality*. The full system (1) describes a seven dimensional problem (3D in space, 3D in velocity and 1D in time), which leads to tremendous cost of grids based methods.

In contrast, the probabilistic methods, which use particles to represent the velocity space, have the advantage of dimension independence. These methods were first studied by Bird [4], known as Direct Simulation Monte Carlo (DSMC) method, for rarefied gas. In plasma simulation, Particle in Cell (PIC) methods [13, 17, 5] are widely used in solving the collisionless Vlasov-Poisson system (i.e. system (1) with  $Q_L = 0$  in the right hand side). The Landau operator (2) is solved by Monte Carlo methods, such as Takizuka-Abe method [28] (henceforth TA) and Nanbu's method [22]. Bobylev and Nanbu [6], Bobylev and Potapenko [7] proposed a general framework in designing particle methods for (2). The PIC-DSMC methods, which combines the PIC methods for the advection part and DSMC methods for the collision part, have been widely used in solving the VPL system (1).

However the particle methods become very inefficient near the fluid regime, where the distribution  $f$  is close to the local equilibrium  $M$ . The reason is that the computation is mainly spent on the collisions between particles sampled from  $M$ . These collisions have no net effect since  $Q_L(M, M) = 0$ . A hybrid method is more favorable in this regime, which decomposes the distribution as

$$f(t, \mathbf{x}, \mathbf{v}) = M(t, \mathbf{x}, \mathbf{v}) + f_d(t, \mathbf{x}, \mathbf{v}). \quad (5)$$

The equilibrium  $M$  is evolved by a fluid solver and the deviation  $f_d$  is solved by particles. Here we allow  $f_d < 0$  and use positive and negative *deviational* particles to represent  $f_d$ .

There have been several related works. In gyrokinetic simulations, the so-called  $\delta f$  methods [8, 21, 16, 14] have been proven to be efficient in collisionless regime. Hybrid methods with weighted deviational particles have also been designed when the collisions are modeled by BGK operator [12] and linear Fokker-Planck operator [27]. However all these methods are very difficult to efficiently

handle binary collisions. Recently Cafilisch et al. [9] and Ricketson et al. [25] studied the thermalization/dethermalization methods with  $f_d(\mathbf{v}) \geq 0$  for the spatial homogeneous Coulomb collisions. However they are difficult to be generalized to the inhomogeneous system (1) without allowing  $f_d$  to be negative. For rarefied gas, Hadjiconstantinou and co-workers [1, 30, 2, 18, 19, 23] have studied a low-variance deviational simulation Monte Carlo (LVDSMC) method in which  $f_d$  could be negative. However, as discussed in Section 2.1.2, the extension to long range Coulomb collisions leads to dramatic growth of particle numbers and hence a numerical instability.

A different Monte Carlo method with deviational particles for Coulomb collisions has been designed by the authors in [31]. By introducing coarse particles as a rough approximation of  $f$ , a large number of collisions are combined efficiently and the growth of particle numbers is reduced from the *numerical* scale to *physical* scale. In this work we seek to extend the method in [31] to the inhomogeneous VPL system (1).

The goal of this work is to design a hybrid method with deviational particles which (a) is applicable in all regimes and (b) is much more efficient than the PIC-DSMC method near the fluid regime. The starting point is an operator splitting between the collision part and the advection part. The collision part is solved as in the homogeneous case cell by cell. As for the advection part, we combine a macro-micro projection method [3] and a Particle-In-Cell method to evolve the macroscopic quantities and all particles. As a result, the number of particles grows in the physical scale and a particle resampling method is applied on both deviational and coarse particles to further control the growth.

The rest of the article is organized as follows. In Section 2 we develop hybrid methods with deviational particles for the VPL system, as well as a simplified system, the VP-BGK system. We describe the algorithms in solving the collision part in Section 2.1 and the advection part in Section 2.2. Next in Section 3 we investigate and improve the particle resampling technique to control the particle growth. Finally in Section 4 various numerical simulations are performed to demonstrate the accuracy and efficiency of our hybrid methods.

## 2 Hybrid Methods with Deviational Particles

As in (5), we decompose the distribution  $f(t, \mathbf{x}, \mathbf{v})$  into a Maxwellian part  $M(t, \mathbf{x}, \mathbf{v})$  and a deviation  $f_d(t, \mathbf{x}, \mathbf{v})$ , which can be both positive and negative.

We use a hybrid representation of  $f$ . A uniform grid is placed in each spatial direction. The Maxwellian part  $M(t, \mathbf{x}, \mathbf{v})$  is given by (3), with the macroscopic quantities defined on the grids.

For simplicity, the Maxwellian part is chosen to have the same the moments of  $f$  as in (4). These moments are solved by a fluid solver.

The deviation part is represented by particles. As  $f_d$  could be both positive and negative, we use *deviational particles* in the representation. In other words, in each spatial cell we sample and evolve *positive* particles for the positive part  $(f_d)_+$  and *negative* particles for the negative part  $(f_d)_-$ . We denote  $N_{p,k}$ ,  $N_{n,k}$  and  $N_{d,k}$  the number of positive particles, negative particles and all deviational particles in the  $k$ th spatial cell  $\mathcal{C}_k$ . Then  $N_{d,k} = N_{p,k} + N_{n,k}$ , for every  $k$ .

The total moments of the deviational particles in each spatial cell are 0, i.e., for  $\phi = 1, \mathbf{v}, |\mathbf{v}|^2/2$ ,

$$\langle \phi f_d(x_k) \rangle = N_{\text{eff}} \sum_{\mathbf{v}_p \in \mathcal{C}_k} \phi(\mathbf{v}_p) - N_{\text{eff}} \sum_{\mathbf{v}_n \in \mathcal{C}_k} \phi(\mathbf{v}_n) = 0, \quad \text{for every } k \text{ and } t. \quad (6)$$

Here in the summation  $\mathbf{v}_p$  are over all positive particles and  $\mathbf{v}_n$  are over all negative particles in each spatial cell  $\mathcal{C}_k$ .  $N_{\text{eff}}$  is the effective number which represents the physical densities per numerical particle. In this work we choose  $N_{\text{eff}}$  to be a prescribed constant for all the time. In other words, we work on the equally weighted particles.

In practice, (6) is only valid in expectation. It cannot hold exactly due to statistical fluctuation unless one enforces the conservations after each step.

To solve the VPL system (1), we start with an operator splitting between the collision part and the advection part. In the following we describe how to evolve the two components  $M$  and  $f_d$  in the two substeps.

## 2.1 Deviational Particles in Collisions

In the substep of collision, the Maxwellian part  $M$  is invariant. Here we describe how to evolve the deviation part  $f_d$  for different collisions.

### 2.1.1 BGK operator

Let us first consider a simplified model called the Vlasov-Poisson-BGK (VP-BGK) system, which is the same as the VPL system (1) except the collision term  $Q_L$  is replaced by the BGK operator,

$$Q_{BGK}(f) = \mu(M - f). \quad (7)$$

This operator models the relaxation of the distribution  $f$  to its local equilibrium  $M$ . Here the collision frequency  $\mu$  is independent of  $\mathbf{v}$ . Inserting the decomposition (5) in the collision substep  $\partial_t f = Q_{BGK}(f)$ , one has

$$\partial_t f_d = -\mu f_d.$$

For this linear model, there are no interactions between particles.

A Monte Carlo method can be designed by applying a forward Euler method,

$$f_d^{n+1} = (1 - \mu\Delta t)f_d^n. \quad (8)$$

Hence each (positive and negative) particle is removed with probability  $\mu\Delta t$  in one time step. Clearly one needs  $\mu\Delta t \leq 1$ .

One can also apply a backward Euler method,

$$f_d^{n+1} = \frac{1}{1 + \mu\Delta t} f_d^n. \quad (9)$$

Again a Monte Carlo method can be designed accordingly. Each (positive and negative) particle is removed with probability  $\frac{\mu\Delta t}{1 + \mu\Delta t}$  in one time step. In this case there is no restriction on the time step  $\Delta t$ .

### 2.1.2 Landau operator

Now we consider the Coulomb collisions modeled by the bilinear Landau operator (2). The binary interaction implies that one has to include all six types of collisions: P-P, P-N, N-N, P-M, N-M and M-M. Here, for example, a P-N collision means a collision between a *positive* particle and a *negative* particle. M means the Maxwellian part. Note that one can omit the M-M collisions which have no net effect. This saves the major cost compared to a full particle method.

To simulate the other five types of collisions, one has two methods: (a) simulate each type of collisions individually; (b) group different types of collisions and simulate them together.

The first method is straightforward. One only needs to determine the numbers of different types of collisions and formulate corresponding collision rules. These rules have been derived in simulating the binary collisions in *rarefied gas* which consists of charge neutral particles (see [1]). They can

be summarized as

$$\begin{aligned}
\text{P-P: } & \mathbf{v}_+, \mathbf{w}_+ \rightarrow \mathbf{v}'_+, \mathbf{w}'_+, \\
\text{P-N: } & \mathbf{v}_+, \mathbf{w}_- \rightarrow 2\mathbf{v}_+, \mathbf{v}'_-, \mathbf{w}'_-, \\
\text{N-N: } & \mathbf{v}_-, \mathbf{w}_- \rightarrow 2\mathbf{v}_-, 2\mathbf{w}_-, \mathbf{v}'_+, \mathbf{w}'_+, \\
\text{P-M: } & M, \mathbf{v}_+ \rightarrow M, \mathbf{w}_-, \mathbf{v}'_+, \mathbf{w}'_+, \\
\text{N-M: } & M, \mathbf{v}_- \rightarrow M, \mathbf{w}_+, \mathbf{v}'_-, \mathbf{w}'_-.
\end{aligned} \tag{10}$$

Here we explain the P-N collision in detail. The other rules can be explained similarly. In a P-N collision between a positive particle  $\mathbf{v}_+$  and a negative particle  $\mathbf{w}_-$ , the two incident particles are removed, four new particles (two positive particles with velocity  $\mathbf{v}$  and two negative particles with velocities  $\mathbf{v}'$  and  $\mathbf{w}'$ ) are created. Here the post-collisional velocities  $\mathbf{v}'$  and  $\mathbf{w}'$  are defined in the usual way in Boltzmann equation (see[10] for example). We refer the reader to [31] for more details on the derivation and implementation of this method.

One can apply these collision rules to the Coulomb gas which consists of charged particles. However a problem of the method based on (10) is that the particle number grows due to collisions in each time step. The increment depends on the number of collisions in one step. For rarefied gas, this problem may be manageable since the number of collisions in one time step is  $\mathcal{O}(\Delta t)$ , hence  $N_d^{t+\Delta t} = (1 + c\Delta t)N_d^t$ . As  $\Delta t \rightarrow 0$ , the number of particles is  $N_d^t = N_d^0 e^{ct}$ , which is  $\Delta t$  independent. This increase is in the *physical* time scale.

However the extension of this method to Coulomb collisions suffers a more severe increase due to the fact that in Coulomb gas all particles are involved in collisions no matter how small the time step  $\Delta t$  is. This is the effect of long range interaction between charged particles. As a result, the particle number increases in the *numerical* time scale, i.e.  $N_d^{t+\Delta t} = cN_d^t$ , with  $c \geq 1$  independent of  $\Delta t$ . Hence the total number of particles increases in a *numerical* scale.

Therefore, instead of simulating each collisions individually, we seek to group different types of collisions and simulate them together to reduce the number of collisions in Coulomb gas. More precisely, in [31] we proposed to split the binary collision operator as

$$Q(f, f) = Q(f, f_d) + Q(f, M) = Q(f, f_d) + Q(f_d, M),$$

and the spatial homogeneous equation  $\partial_t f = Q(f, f)$  becomes

$$\begin{cases} \partial_t M = 0, \\ \partial_t f_d = Q(f, f_d) + Q(f_d, M). \end{cases} \tag{11}$$

Here the operator  $Q(f, f_d)$  is defined to be asymmetric in  $f$  and  $f_d$ , so that  $Q(f, f_d)$  represents the changes in  $f_d$  due to collisions with  $f$ . Note that  $Q(f, f_d) = Q(f, (f_d)_+) - Q(f, (f_d)_-)$ . One has grouped M-P, P-P and N-P collisions in the first term  $Q(f, (f_d)_+)$  since

$$Q(f, (f_d)_+) = Q(M, (f_d)_+) + Q((f_d)_+, (f_d)_+) - Q((f_d)_-, (f_d)_+).$$

Similarly M-N, P-N and N-N collisions have been grouped and represented by the term  $Q(f, (f_d)_-)$ .

To solve the term  $Q(f, f_d)$ , one performs regular collisions between particles sampled from  $f$  and the deviational particles. However, sampling from  $f$  at every time step is very expensive, since one needs to recover the distribution  $(f - M)$  in 3D velocity space from the deviational particles. Instead of sampling from  $f$ , we proposed in [31] to include a bunch of *coarse particles* which solve the original homogeneous equation  $\partial_t f = Q(f, f)$ . To sample a particle from  $f$ , one only needs to randomly pick up a coarse particle. More specifically, we proposed to solve the following system

$$\begin{cases} \partial_t f_c = Q(f_c, f_c), \\ \partial_t M = 0, \\ \partial_t f_d = Q(f_c, f_d) + Q(f_d, M). \end{cases} \tag{12}$$

Here  $f_c$  solves the original homogeneous Landau equation. However,  $f_c$  serves as a coarse approximation of the solution and is represented by a relatively small number of particles, i.e. the *coarse particles*. One can evolve the coarse particles by a regular Monte Carlo method, for example the TA method or Nanbu's method.

Now the evolution of the deviational particles involve two steps. First, to solve the  $Q(f_c, f_d)$  part, the deviational particles are updated by performing regular collisions with the coarse particles. In this step  $N_d$ , the number of deviational particles, does not change. Then, a number of  $\mathcal{O}(\Delta t N_d)$  deviational particles are sampled from the distribution  $Q(f_d, M)$  and added into the system. Therefore after one step of collision,  $N_d$  increases by  $N_d^{t+\Delta t} = (1 + c\Delta t)N_d^t$ , i.e., in the *physical* time scale.

The Maxwellian part  $M$  does not change in the collision step. Therefore the coarse particles provide a coarse, direct simulation of  $f$ , while the deviational particles give a fine, deviational simulation of  $(f - M)$ . This method can be applied to any type of binary collisions, for example, the short range hard sphere collision and the long range Coulomb collision.

To perform the collisions between the deviational particles and the coarse particles, i.e. the  $Q(f_c, f_d)$  part, the number of coarse particles should be large enough. One needs

$$N_{c,k} \geq N_{d,k}, \quad \text{for every } k, \quad (13)$$

with  $N_{c,k}$  the number of coarse particles in the spatial cell  $\mathcal{C}_k$ . We assume all the coarse particles in all cells carry the same weight to avoid the collisions between particles with variable weights. However, the effective number of the coarse particles could be time dependent as  $N_d$  varies in time. We will discuss more on this in Section 3.2.

Finally, note that each new sampled particle needs both spatial and velocity coordinates. To sample a particle with velocity distribution to be  $Q(f_d, M)$ , we refer the reader to Section 4 in [31]. The spatial coordinate of each sample is set to be uniformly distributed in current spatial cell  $\mathcal{C}_k$ .

**Remark 2.1.** The system (12) describes an particle method for a general bilinear operator  $Q(f, f)$  which admits an equilibrium  $M$ , i.e.,  $Q(M, M) = 0$ . One can formally apply it to Landau operator (2). However, the existing Monte Carlo methods for Coulomb collisions such as Takizuka-Abe method [28] and Nanbu's method [22] are not solving the Landau equation  $\partial_t f = Q_L(f, f)$  directly. They instead solve the formula proposed in [6],

$$f(\mathbf{v}, t + \Delta t) = \frac{1}{\rho} \iint_{\mathbb{R}^3 \times \mathbb{S}^2} D\left(\frac{\mathbf{u} \cdot \mathbf{n}}{u}, A \frac{\Delta t}{u^3}\right) f(\mathbf{w}', t) f(\mathbf{v}', t) d\mathbf{w} d\mathbf{n}, \quad (14)$$

with  $D(\cdot, \cdot)$  an approximated kernel determined by the numerical algorithms,  $\mathbf{u} = \mathbf{v} - \mathbf{w}$ ,  $\rho$  the density of  $f$  and  $A$  the coefficient in (2). This is a first order approximation of the solution  $f(t^{n+1})$ . Therefore, in practice one applies the method (12) to (14), leading to a first order approximation of (12),

$$\begin{cases} f_c(\mathbf{v}, t + \Delta t) = \frac{1}{\rho} \iint D f_c(\mathbf{w}') f_c(\mathbf{v}') d\mathbf{w} d\mathbf{n}, \\ M(\mathbf{v}, t + \Delta t) = M(\mathbf{v}, t), \\ f_d(\mathbf{v}, t + \Delta t) = \frac{1}{\rho} \iint D f_c(\mathbf{w}') f_d(\mathbf{v}') d\mathbf{w} d\mathbf{n} + \Delta M(\mathbf{v}), \end{cases} \quad (15)$$

where

$$\Delta M(\mathbf{v}) = \frac{1}{\rho} \iint D f_d(\mathbf{w}') M(\mathbf{v}') d\mathbf{w} d\mathbf{n} - \frac{\rho_p - \rho_n}{\rho} M(\mathbf{v}) \quad (16)$$

We refer the reader to Section 4 in [31] for details.

## 2.2 Deviational Particles in Advection

Now consider the advection part, which is commonly known as the collisionless Vlasov-Poisson equation,

$$\partial_t f + \mathbf{v} \cdot \nabla_{\mathbf{x}} f - \mathbf{E} \cdot \nabla_{\mathbf{v}} f = 0, \quad (17)$$

with the electric field  $\mathbf{E}$  given by the Poisson equation (1b), where the density  $\rho$  is computed from

$$\rho(x_k) = \rho_M(x_k) + \frac{N_{\text{eff}}}{|\mathcal{C}_k|} (N_{p,k} - N_{n,k})$$

in each spatial cell  $\mathcal{C}_k$ . Here  $\rho_M$  is the density of the Maxwellian part,  $N_{p,k}$  and  $N_{n,k}$  are the numbers of positive and negative deviational particles in that cell.  $|\mathcal{C}_k|$  is the size of the cell. Note that  $N_{p,k}$  and  $N_{n,k}$  might not be exactly the same due to the statistical fluctuation.

In our method,  $\rho_M$ ,  $N_p$  and  $N_n$  are computed explicitly from  $M$  and deviational particles at the beginning of the advection step. Then  $\mathbf{E}$  is computed from the Poisson equation (1b). Denote

$$\mathcal{T} = \mathbf{v} \cdot \nabla_{\mathbf{x}} - \mathbf{E} \cdot \nabla_{\mathbf{v}}.$$

Now  $\mathcal{T}$  is a linear operator during current time step.

In the advection step, we need to evolve both the Maxwellian part  $M$  and the deviational particles which represent  $f_d$ . For the Coulomb collisions, we also need to evolve the coarse particles which represent  $f_c$ .

For  $M$  and  $f_d$ , we employ the macro-micro decomposition introduced in [3]. First (17) is reformulated as

$$\begin{cases} \frac{\partial}{\partial t} \langle M \phi \rangle + \nabla_{\mathbf{x}} \cdot \langle \mathbf{v} M \phi \rangle + \nabla_{\mathbf{x}} \cdot \langle \mathbf{v} f_d \phi \rangle = \langle \mathbf{E} \cdot \nabla_{\mathbf{v}} f \phi \rangle = (0, -\rho \mathbf{E}, -\rho \mathbf{u} \cdot \mathbf{E})^T, \\ \frac{\partial f_d}{\partial t} + \mathcal{T} f_d = - \left( \frac{\partial}{\partial t} + \mathcal{T} \right) M, \end{cases} \quad (18)$$

with  $\phi = 1, \mathbf{v}, |\mathbf{v}|^2/2$ . Here the first equation is obtained by taking the moments of both sides of (17), while the second equation is just a rearrangement of (17).

To solve the first equation, one can apply a fluid solver to compute the term  $\nabla_{\mathbf{x}} \cdot \langle \mathbf{v} M \phi \rangle$ . For the next term one can use the approximation

$$\langle \mathbf{v} f_d \phi \rangle|_{x_k} \approx N_{\text{eff}} \sum_{\mathbf{v}_p \in \mathcal{C}_k} (\mathbf{v}_p \phi(\mathbf{v}_p)) - N_{\text{eff}} \sum_{\mathbf{v}_n \in \mathcal{C}_k} (\mathbf{v}_n \phi(\mathbf{v}_n)). \quad (19)$$

For the second equation of (18), to eliminate the time derivative term  $\partial_t M$ , we define a projection operator  $\Pi_M$  by

$$\Pi_M \psi = \frac{M}{\rho_M} \left[ \langle \psi \rangle + \frac{(\mathbf{v} - \mathbf{u}_M) \cdot \langle (\mathbf{v} - \mathbf{u}_M) \psi \rangle}{T_M} + \frac{1}{2d} \left( \frac{|\mathbf{v} - \mathbf{u}_M|^2}{T_M} - d \right) \left\langle \left( \frac{|\mathbf{v} - \mathbf{u}_M|^2}{T_M} - d \right) \psi \right\rangle \right], \quad (20)$$

where  $\rho_M$ ,  $\mathbf{u}_M$  and  $T_M$  are the density, macroscopic velocity and temperature of  $M$ .  $d$  is the dimension of velocity space. We take  $d = 3$  in this work.

From (6), one has

$$\Pi_M f_d = \Pi_M (\partial_t f_d) = (I - \Pi_M) (\partial_t M) = 0.$$

Apply the operator  $(I - \Pi_M)$  on both sides of the second equation of (18), one obtains

$$\frac{\partial f_d}{\partial t} + \mathcal{T} f_d = -(I - \Pi_M) (\mathcal{T} M) + \Pi_M (\mathcal{T} f_d).$$



This equation governs the evolution of the deviational particles. There are two processes involved: the advection of particles and the sampling from the source term on the right hand side. The advection process is just like a regular PIC method: one moves all the deviational particles by solving the ODE system

$$\dot{\mathbf{x}} = \mathbf{v}, \quad \dot{\mathbf{v}} = -\mathbf{E}, \quad (21)$$

with  $\mathbf{E}$  the electric field in the spatial cell the particle resides. In this process  $N_d$  is not changed. Then in the sampling process, deviational particles are created due to the fact that  $M$  is driven away from the shape of a Maxwellian after convected. Noting that  $f_d$  is not included in the first term, a number of  $\mathcal{O}(\Delta t (N_{\text{eff}}^{-1} + N_d))$  particles are created. The deviational particles are created even if they do not exist at beginning. After created,  $N_d$  increases in the *physical* time scale, just as in the Coulomb collision step (12). In Appendix A we give an efficient algorithm to sample velocities from the source term.

Again the spatial coordinate of each new sampled particle is set to be uniformly distributed in current spatial cell  $\mathcal{C}_k$ .

Furthermore, note that the moments of the deviational particles remain zero in expectation. This is due to the fact that, for any distribution  $\psi$ ,

$$\langle \phi(I - \Pi_M)\psi \rangle = 0, \quad \text{for } \phi = 1, \mathbf{v}, |\mathbf{v}|^2/2.$$

Finally, for the VPL system, one also needs to transport the coarse particles which represent  $f_c$  in (12). Similar to the collision step,  $f_c$  just solves the original Vlasov-Poisson equation (17) in the advection step. One can simply evolve the coarse particles by a regular PIC method.

Therefore the system to be solved can be summarized as,

$$\frac{\partial f_c}{\partial t} + \mathcal{T}f_c = 0, \quad (22a)$$

$$\frac{\partial}{\partial t} \langle M\phi \rangle + \nabla_{\mathbf{x}} \cdot \langle \mathbf{v}M\phi \rangle + \nabla_{\mathbf{x}} \cdot \langle \mathbf{v}f_d\phi \rangle = (0, -\rho\mathbf{E}, -\rho\mathbf{u} \cdot \mathbf{E})^T, \quad (22b)$$

$$\frac{\partial f_d}{\partial t} + \mathcal{T}f_d = -(I - \Pi_M)(\mathcal{T}M) + \Pi_M(\mathcal{T}f_d). \quad (22c)$$

**Remark 2.2.** The advection of deviational particles was also studied in [19]. In that work, the Maxwellian part  $M$  is assumed to be invariant during the advection time step. The time evolution of the macroscopic quantities are left in the collision part. Hence in the advection step one only updates the deviational particles, by solving the second equation in (18) with  $\frac{\partial}{\partial t}M = 0$ . The macro-micro projection was not used.

### 2.3 Summarize

We refer to the Hybrid methods with Deviational Particles developed in this section as the HDP methods. Now we summarize the HDP methods for VP-BGK system and VPL system.

First, for both systems, an operator splitting is applied between the advection part and the collision part. Then different algorithms are applied in each substep.

#### HDP method for VP-BGK system.

- In the collision substep, one solves (8) or (9).
  - The Maxwellian part  $M$  keeps invariant.
  - The deviational particles are removed with rates  $\mu\Delta t$  or  $\frac{\mu\Delta t}{1+\mu\Delta t}$ .
- In the advection substep, one solves (22b) and (22c).



- The moments of the Maxwellian part  $M$  are updated by applying a fluid solver on (22b).
- The deviational particles move in the phase space by solving (21). New deviational particles sampled from the right hand side of (22c) are added into the system.

**HDP method for VPL system.** Coarse particles are needed to solve this system.

- In the collision step, one solves (12).
  - The coarse particles are updated by applying a regular DSMC method to simulate the Coulomb collision.
  - The Maxwellian part  $M$  keeps invariant.
  - The deviational particles collide with the coarse particles. New deviational particles sampled from  $\Delta t Q(f_d, M)$  are added into the system.
- In the advection step, one solves (22).
  - The coarse particles move in the phase space by solving (21). This is the traditional PIC method.
  - The moments of the Maxwellian part  $M$  are updated by applying a fluid solver on (22b).
  - The deviational particles move in the phase space by solving (21). New deviational particles sampled from the right hand side of (22c) are added into the system.

**Remark 2.3.** In the VP-BGK system, the collision step automatically reduces  $N_d$  by a constant ratio in one time step. As a result, for a moderate collision frequency  $\mu$ , the collision step can stabilize the advection step. Furthermore, the positive and negative deviational particles cannot be mixed in neither the collision nor the advection step.

In the VPL system, the number of deviational particles increases by  $\mathcal{O}(\Delta t N_d)$  in both advection step and the collision step. Hence  $N_d$  grows exponentially in the physical time scale. However, as will be discussed in next section, the Coulomb collision can mix the positive and negative deviational particles, making it possible to reduce  $N_d$  by canceling particle pairs if they are close in the phase space.

### 3 Control of Particle Growth

As discussed in Remark 2.3, the number of deviational particle grows exponentially in the physical scale when solving the VPL system (1) by the HDP method. A possible way to control the growth is to remove particle pairs if they are close in the phase space. Similar to the homogeneous case studied in [31], it is both necessary and efficient to apply a particle reduction technique.

- First, to perform the collisions between the deviational particles and the coarse particles, i.e. the  $Q(f_c, f_d)$  term in (12), one needs (13) to be valid in each cell. However, the coarse particles are evolved by applying a regular DSMC method on the collisions and a regular PIC method on the advection. Hence  $N_c$  does not change while  $N_d$  increases in the physical scale. After a moderate time period, the assumption (13) could fail. Therefore it is necessary to reduce  $N_d$  periodically.
- Second, the first term  $Q(f, f_d)$  in the collision step drives both  $(f_d)_+$  and  $(f_d)_-$  to the same distribution. The positive and negative deviational particles are approaching each other. Therefore by canceling particle pairs which are close in phase space, one can reduce the number of particles and expect  $N_d$  to be bounded and even reduced. This increases the efficiency of the whole method.

### 3.1 Resample Deviational Particles

A particle resampling method was studied in [31]. One first finds an approximation of the explicit formula of the deviation part  $f_d$  and then resamples deviational particles from it. More specifically,  $f_d$  is approximated by

$$\begin{aligned} f_d(\mathbf{v}) &= \sum_{\mathbf{k}} \langle f_d, \phi_{\mathbf{k}} \rangle \phi_{\mathbf{k}}(\mathbf{v}) \approx \sum_{|\mathbf{k}| < K} \langle f_d, \phi_{\mathbf{k}} \rangle \phi_{\mathbf{k}}(\mathbf{v}) \\ &\approx N_{\text{eff}} \sum_{|\mathbf{k}| < K} \left( \sum_{\mathbf{v}_p} \phi_{\mathbf{k}}(\mathbf{v}_p) - \sum_{\mathbf{v}_n} \phi_{\mathbf{k}}(\mathbf{v}_n) \right) \phi_{\mathbf{k}}(\mathbf{v}). \end{aligned} \quad (23)$$

Here  $\{\phi_{\mathbf{k}}\}$  is a complete basis set.  $\langle \cdot, \cdot \rangle$  is the inner product, and  $K$  is a cutoff parameter.

The total cost is  $\mathcal{O}(N_d K^3)$  to evaluate the coefficients in (23) and to resample new particles. We find at least  $K = 20$  is needed to make a resampling with reasonable accuracy. This makes the cost to be  $\mathcal{O}(8000 N_d)$  which is pretty large. This cost is acceptable in the spatial homogeneous case, since the distribution decays to equilibrium monotonically and the number of deviational particles becomes very small as time evolves. But in the spatial inhomogeneous case, the advection part also creates a lot of particles and drives the distribution away from the Maxwellian. A lot more resamplings are needed. To accelerate the resampling, in Appendix B we propose an approximation of the formula (23), which reduces the total cost to  $\mathcal{O}(N_d + K^3 \log K)$ .

In the spatial homogeneous case, the resampling is performed when the condition (13) is violated. So a straightforward algorithm for the inhomogeneous system is that, one performs the resampling in the cell where (13) is violated. However this strategy is not very efficient. Consider the case that the condition (13) is violated and the resampling is performed in some cell for the first time. Then the neighboring spatial cells, in which no resampling has been performed and hence many deviational particles are accumulated, might convect many particle pairs into this cell in the next time step. This makes the previous resampling meaningless.

Here we use the following strategy. Whenever the condition (13) is violated in one cell, the resampling of deviational particles is performed in all the cells  $\mathcal{C}_k$  where

$$N_{d,k} \geq \beta N_{c,k}, \quad (24)$$

with a prescribed constant  $\beta < 1$ .

The case  $\beta = 1$  is exactly the algorithm that the resampling is performed only in the cells violating (13). If  $\beta$  is very small, many unnecessary resamplings will be performed in the cells where  $N_{d,k} \ll N_{c,k}$ . In practice we find a simple choice  $\beta = 0.9$  gives quite satisfactory efficiency.

**Remark 3.1.** In the collision process, the two types of particles are mixed due to the collisions with the coarse particles. However there is no such effect in the advection step. Once the electric field is computed, the advection operator is linear and time invertible. The supports of the positive and negative particles keep disjoint during the advection, if they are disjoint initially. Therefore to make effective resampling, the collision cannot be weak. This is also the starting point of using deviational particles, i.e. we assume the distribution is close to the local equilibrium.

**Remark 3.2.** For the spatial inhomogeneous case, we need to apply the particle resampling in each spatial cell, instead of the whole  $(\mathbf{x}, \mathbf{v})$  phase space, due to two reasons. First, the error introduced by the cutoff after  $K$  terms in (23) depends on the smoothness of the underlying function  $f_d$ . In velocity direction  $f_d$  is smooth due to the Coulomb collisions (see [26]). However  $f_d$  might be discontinuous in space coordinates due to boundary conditions or external forces. Second, resampling in the whole  $(\mathbf{x}, \mathbf{v})$  phase space leads to a memory usage of  $\mathcal{O}(K^3 N_x^3)$  to store the coefficients in (23). This loses the advantages of using particles.

## 3.2 Resample Coarse Particles

The coarse particles also need resampling, which means changing the effective number of the coarse particles. This effective number, denoted by  $N_{\text{eff}}^c$ , could be both increased and decreased by removing or adding some coarse particles. Noting that all the coarse particles in all cells carry the equal weight, we have to perform resampling in all cells once it is needed in one cell.

Let  $N_c$  be the total number of coarse particles. There are three cases one needs to resample the coarse particles.

### Case 1. Reduce $N_c$ to improve efficiency.

We first recall the resampling of coarse particles in the spatial homogeneous case. Note that  $(f_d)_+$  and  $(f_d)_-$ , the positive and negative parts of the deviation, approach the same equilibrium as time evolves. Hence  $N_d$ , the number of deviational particles, is reduced significantly after each particle resampling. As the only requirement on the coarse particle is (13), one can resample the coarse particles by simply discarding a fraction of them. In this case the effective number  $N_{\text{eff}}^c$  increases. Eventually only a small number of deviational and coarse particles are needed.

However this is not true in the spatial inhomogeneous case. The advection part drives the distribution away from the local equilibrium. The balance between the advection part and the collision part keeps  $N_d$  from being diminished.  $N_d$  is not likely to be reduced significantly as in the homogeneous case. Since the resampling of the coarse particles always involves all the cells, in this work we choose *not* to reduce  $N_c$  even if  $N_d$  is reduced through resampling in some cell.

### Case 2. Increase $N_c$ if the resampling of deviational particles fails.

The resampling of deviational particles may fail in some cell if the growth of  $N_d$  is mainly due to the advection part. This could happen, for example, in the early stage of simulation with initial distribution to be local equilibrium, or in the regime with large spatial variation. In these cases, the overlap between the positive and negative deviational particles is very small, leading to very small and even no reduction of  $N_d$  in a resampling. To satisfy (13), we need to introduce more coarse particles.

However, one cannot obtain more *independent* coarse particles from only the original coarse particles. To increase  $N_c$  effectively, one needs to sample directly from  $f = M + f_d$ , with  $f_d$  recovered from the deviational particles as in (23) or in Appendix B. One needs to sample more coarse particles in all cells to ensure that their effective numbers remain the same.

### Case 3. To conform to the fine solution.

After simulation of a relatively long time without resampling coarse particles, the accumulated error in the coarse solution  $f_c$  might become significant. In this case one can discard the original coarse particles and resample them from  $f = M + f_d$  directly. This keeps the coarse solution consistent with the fine solution.

In practice, this resampling includes two steps. First, we resample the deviational particles in all cells. Then coarse particles are resampled with the new effective number

$$(N_{\text{eff}}^c)_{\text{new}} = \max \left( \min_k \frac{\rho_M(x_k)|\mathcal{C}_k|}{\gamma N_{d,k}}, (N_{\text{eff}}^c)_{\text{old}} \right), \quad (25)$$

with  $\gamma \gtrsim 1$  a prescribed constant. We take  $\gamma = 1.1$  in the simulations. This resamples no more than the old coarse particles without violating the condition (13) in each cell.

As a result, the resampling of coarse particles in case 2 increases  $N_c$ . However, as time evolves the distribution  $f$  might approach some global equilibrium, which demands less deviational particles. The resampling in case 3 has a chance to reduce the number of coarse particles.

### 3.3 Assign x coordinates after resampling

Finally, one needs to assign an  $x$  coordinate to each resampled particle. A simple choice is to pick up a random number uniformly distributed in the spatial cell as the new coordinate. This introduces a first order error in cell size  $\Delta x$ . It is possible to obtain a second order reconstruction by approximating the coefficients in the expansion (23) by linear functions in  $x$ . This will be further investigated in future work.

### 3.4 Another viewpoint of the coarse particles

The evolution and resampling of coarse particles can be explained in another way. As discussed in Section 2.1.2, sampling from  $f$  at every time step to solve (11) is very expensive. Hence once particles are sampled from  $f = M + f_d$ , they are not discarded immediately after that time step. Instead these particles are evolved and reused in the following time steps. After a while, when these particles cannot represent  $f$  accurately due to the error accumulation, we discard these particles and sample a new collection of particles from  $f = M + f_d$ . This is exactly the evolution and resampling of coarse particles.

## 4 Numerical Tests

In the following we perform numerical simulations on the VP-BGK system and the VPL system (1). The HDP methods are applicable to phase spaces with any dimensions. Here for simplicity our tests are performed on the 1D spatial and 3D velocity coordinates.

In the tests, we take spatial domain to be  $x \in [0, 4\pi]$ , with a periodic boundary condition in  $x$ . Note that a truncation in velocity space is not necessary since we use particles to represent the distribution in  $\mathbf{v}$ . In our tests, we take the coefficient  $A = 10$  in (2). Unless specified, we always use  $N_x = 400$  grid points in  $x$  direction.

We update the macroscopic quantities in the Maxwellian by applying a kinetic scheme (see [11, 24] for example) on (22b). As this equation is solved in an explicit deterministic way, the CFL condition is needed. In the following simulations we find the macroscopic velocity is always less than 5. So we always take  $\Delta t = \Delta x/10$  except in Section 4.2.2. The restriction on the time step due to electric field is  $\mathcal{O}(1)$ , hence satisfied automatically.

The simulations are based on the Landau damping problem. The initial distribution is at equilibrium  $f(t=0, x, \mathbf{v}) = M(t=0, x, \mathbf{v})$  with the macroscopic quantities given by

$$\begin{cases} \rho(t=0, x) = 1 + \alpha \sin(x), \\ \mathbf{u}(t=0, x) = 0, \\ T(t=0, x) = 1, \end{cases} \quad (26)$$

where  $\alpha$  is a constant characterizing how close the initial distribution is to the global equilibrium. Both the linear and nonlinear Landau damping are considered in the following tests.

### 4.1 HDP method for VP-BGK system

In the first test, our aim is to check if our methods can capture the Landau damping properties of the VP-BGK system. In this case the coarse particles are not needed. Thanks to the simple algorithms (8) or (9) on the collision, this model allows us to focus on the advection substep.

The reference solution is solved by a PIC-DSMC method. The advection is solved by a regular PIC method. The algorithms of Monte Carlo Collision on the homogeneous BGK equation  $\partial_t f = Q_{BGK}(f)$  are obtained by applying a forward Euler method

$$f^{n+1} = \Delta t \mu M + (1 - \Delta t \mu) f^n, \quad (27)$$

or a backward Euler method

$$f^{n+1} = \frac{1}{1 + \Delta t \mu} M + \frac{\Delta t \mu}{1 + \Delta t \mu} f^n. \quad (28)$$

Hence a fraction of  $\Delta t \mu$  or  $\frac{1}{1 + \Delta t \mu}$  particles are removed and replaced by particles sampled from  $M$ . Again one needs  $\Delta t \mu \leq 1$  in the explicit method (27).

In the following the explicit methods (8) and (27) are applied. We use the same effective number  $N_{\text{eff}} = 5 \times 10^{-7}$  in both the HDP method and the PIC-DSMC method.

In Figure 1 and 2 a linear Landau damping problem is studied, with  $\alpha = 0.01$  in (26). A weak collision frequency  $\mu = 1$  is used. The top of Figure 1 compares the damping of electric field

$$||E||^2 = \Delta x \sum_{k=1}^{N_x} (E_k)^2 \quad (29)$$

between the solutions obtained by HDP method (red dots) and the PIC-DSMC method (black lines). A good agreement is observed. In the bottom figure, the time evolution of  $N_p$  and  $N_n$  (i.e. the numbers of positive and negative components of the deviational particles) in HDP method are plotted. The two lines overlap each other, suggesting that the deviation part does not carry any mass. The growth of particles caused by the advection part gets balanced by the collision as time evolves. For comparison,  $2.5 \times 10^7$  particles are used in the PIC-DSMC methods. The HDP methods uses less than 3% particles. Figure 2 provides a snapshot of the solutions in the  $(x, v_1)$  phase space obtained by HDP method and PIC-DSMC method at time  $t = 1.25$ . The top three figures correspond to  $M$ ,  $(f_d)_+$  and  $(f_d)_-$  in the HDP solution. The bottom figures correspond to the distribution  $f = M + f_d$  in the HDP solution (left) and  $f$  in the PIC-DSMC solution (right). In this case the deviation  $f_d$  is very small and the solutions obtained in two methods agree well with each other.

In Figure 3 and 4 a nonlinear Landau damping problem is studied, with  $\alpha = 0.4$  in (26). Again a weak collision frequency  $\mu = 1$  is used. In this case one can observe the decay of particle numbers as time evolves, which indicates the solution is approaching a global equilibrium which has a uniform density in space. The snapshot in Figure 4 clearly shows the deviation of the solution to the local equilibrium. Again the solutions obtained by the two methods agree well with each other.

In Figure 5 and 6 the same nonlinear Landau damping problem is studied with a strong collision frequency  $\mu = 10$ . Now one can observe the oscillation in the particle number, which illustrates the competition between the advection part and the collision part.  $N_d$  gradually decays in the long time, which suggests the decay of the solution to the global equilibrium. The deviation part is smaller than the weak collision case and fewer particles are needed.

## 4.2 HDP method for VPL system

Now we apply the HDP method on the VPL system (1) with the same initial data (26). Again the reference solution is solved by a PIC-DSMC method, with a PIC method on the advection, and a DSMC method by Takizuka-Abe [28] on the Coulomb collisions. In this case we have to include the coarse particles and perform the particle resampling whenever needed.

### 4.2.1 Landau Damping

Figure 7 presents the results on the linear Landau damping problem with  $\alpha = 0.01$ . The top figure shows the good agreement between the solution obtained from the HDP method and the reference solution by a PIC-DSMC method. The bottom figure gives the time evolution of the two components of  $N_d$  (the number of deviational particles) and  $N_c$  (the number of coarse particles). The zigzag pattern in  $N_d$  results from the resampling of coarse particles. In the early stage, one can

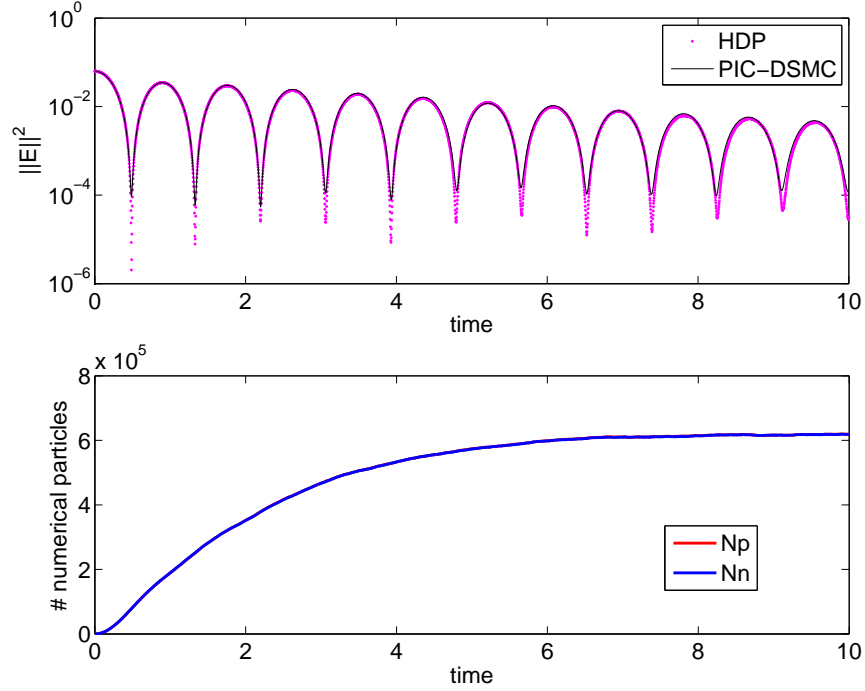


Figure 1: The linear Landau damping problem of the VP-BGK system with collision rate  $\mu = 1$ . Top: the comparison of the decay of energy between the HDP method (red dots) and PIC-DSMC method (black line). Bottom: the growth of the number of positive and negative deviational particles. In the bottom figure, the two lines overlap each other.

see the growth of  $N_c$  when the resampling of deviational particles fails, which is the case 2 discussed in Section 3.2. After the transient time period,  $N_d$  and  $N_c$  stop from growing. The reduction of  $N_c$  from resampling can be observed, which is the case 3 in Section 3.2.

Figure 8 shows the snapshots of the solutions from the HDP method and the PIC-DSMC method at time  $t = 1.25$ . Again, due to the small number of deviational particles, the deviation part  $f_d$  is almost invisible. The solutions are close to a global equilibrium.

Figure 9 tests on the nonlinear Landau damping problem with  $\alpha = 0.4$ . Again, the top figure illustrates the good agreement between the solution from the HDP method and the reference solution. The bottom figure shows the time evolution of  $N_d$  and  $N_c$ . A similar pattern on the growth of the particle numbers is observed as in the linear Landau damping problem.

Figure 10 gives the snapshots of the solutions from the HDP method and the PIC-DSMC method at time  $t = 1.25$ . Now the deviation part is significant. The vertical lines in  $(f_d)_+$  and  $(f_d)_-$  are resulted from the resampling of deviational particles in those cells.

The aim of this simulation is to show that the HDP method is applicable in all regimes. But we would like to mention that the HDP method for nonlinear Landau damping is not more efficient than the PIC-DSMC method since the deviation part  $f_d$  is quite large and many deviational particles (and hence coarse particles) are needed.

#### 4.2.2 Convergence test

Now we perform a numerical test to check convergence of the HDP method on the VPL system in all regimes, as the effective number  $N_{\text{eff}} \rightarrow 0$  (or equivalently particle number  $N_d \rightarrow \infty$ ), and as the time step  $\Delta t \rightarrow 0$ .

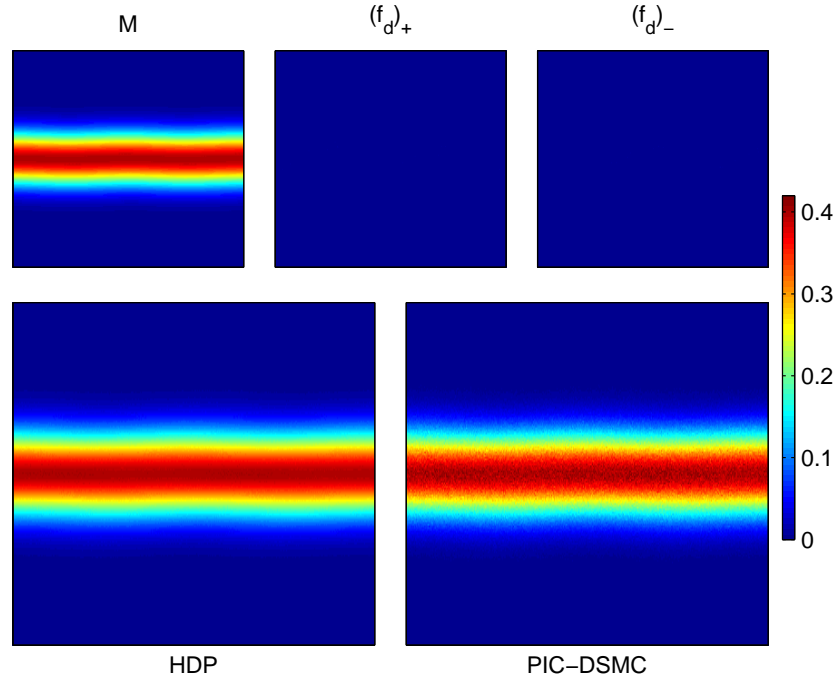


Figure 2: The snapshots of the distribution in the  $x - v_1$  phase space at time  $t = 1.25$  in the linear Landau damping problem of the VP-BGK system with  $\mu = 1$ . The top three figures show the components  $M$ ,  $(f_d)_+$  and  $(f_d)_-$  in the HDP method. The bottom left figure gives the solution  $M + f_d$  obtained by HDP method. The bottom right gives the solution  $f$  obtained by PIC-DSMC method.



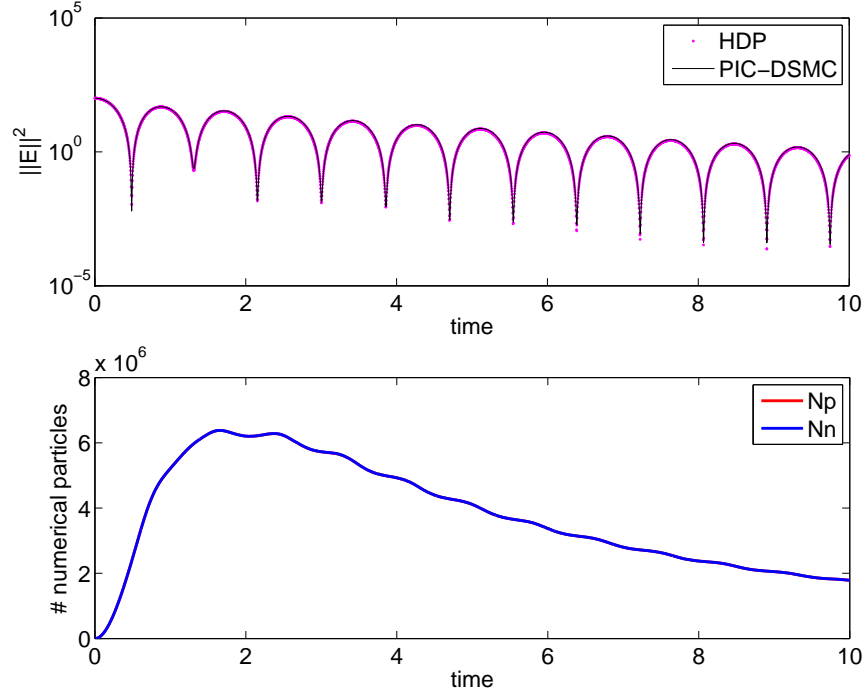


Figure 3: The nonlinear Landau damping problem of the VP-BGK system with collision rate  $\mu = 1$ . Top: the comparison of the decay of energy between the HDP method (red dots) and PIC-DSMC method (black line). Bottom: the growth of the number of positive and negative deviational particles. In the bottom figure, the two lines overlap each other.

We consider the relative error in the distribution

$$\text{error} = \frac{\|f - f_{\text{ref}}\|_1}{\|f\|_1}, \quad (30)$$

with  $\|\cdot\|_1$  the  $L^1$  norm.  $f_{\text{ref}}$  is the reference solution solved by the HDP method with a much smaller time step and effective number.

The convergence test as  $N_{\text{eff}} \rightarrow 0$  is performed on the nonlinear Landau damping problem with  $\alpha = 0.4$ .  $\Delta x$  and  $\Delta t$  are fixed. We take  $N_{\text{eff}}$  ranging from  $10^{-5}$  to  $1.25 \times 10^{-6}$ . The reference solution is obtained with  $N_{\text{eff}} = 6.25 \times 10^{-7}$ . The coarse particles use the same effective number. As illustrated in the left of Figure 11, a half order convergence is observed in both the coarse solution, which is solved by a PIC-DSMC method, and the fine solution, which is solved by the HDP method.

The convergence test as  $\Delta t \rightarrow 0$  is performed on the linear Landau damping problem with  $\alpha = 0.01$ . In this case the effective number is fixed to be  $N_{\text{eff}} = 10^{-9}$ . As shown in the right of Figure 11, the HDP method has a first order convergence in  $\Delta t$ . As for the coarse solution solved by the PIC-DSMC method, we find the error in time step  $\Delta t$  is completely dominated by the statistical error due to finite number of samples. To observe the first order time step error, the size of samples becomes too large to be simulated. The comparison between the two figures in Figure 11 also suggests that the deterministic error (due to time step  $\Delta t$ ) is much smaller than the statistical error (due to particle number  $N$ ).

#### 4.2.3 Efficiency test

Finally we perform an efficiency test on the HDP methods for the VPL system. The HDP methods are designed to accelerate the simulation of PIC-DSMC methods in the near fluid regime. To test

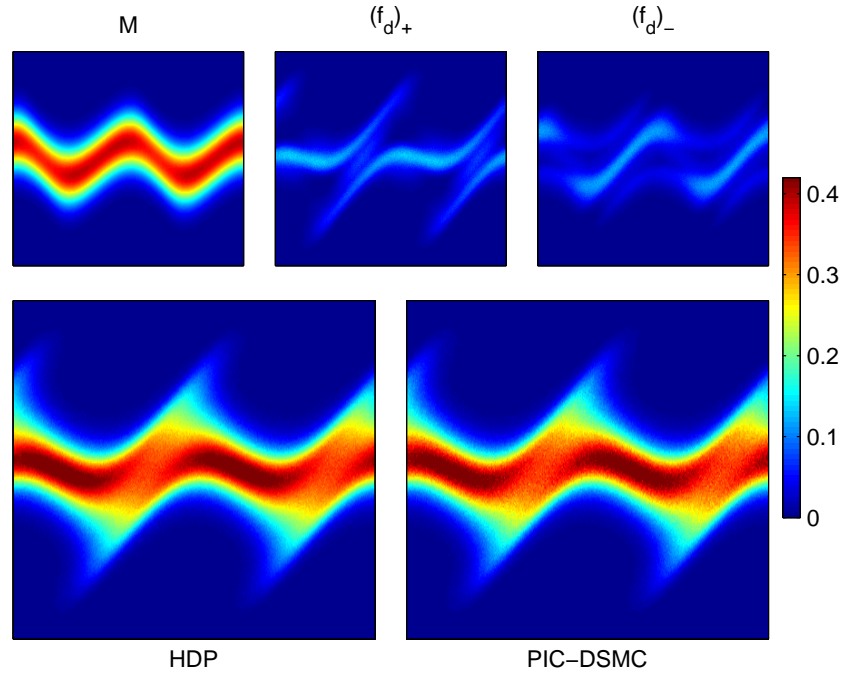


Figure 4: The snapshots of the distribution in the  $x - v_1$  phase space at time  $t = 1.25$  in the nonlinear Landau damping problem of the VP-BGK system with  $\mu = 1$ . The top three figures show the components  $M$ ,  $(f_d)_+$  and  $(f_d)_-$  in the HDP method. The bottom left figure gives the solution  $M + f_d$  obtained by HDP method. The bottom right gives the solution  $f$  obtained by PIC-DSMC method.

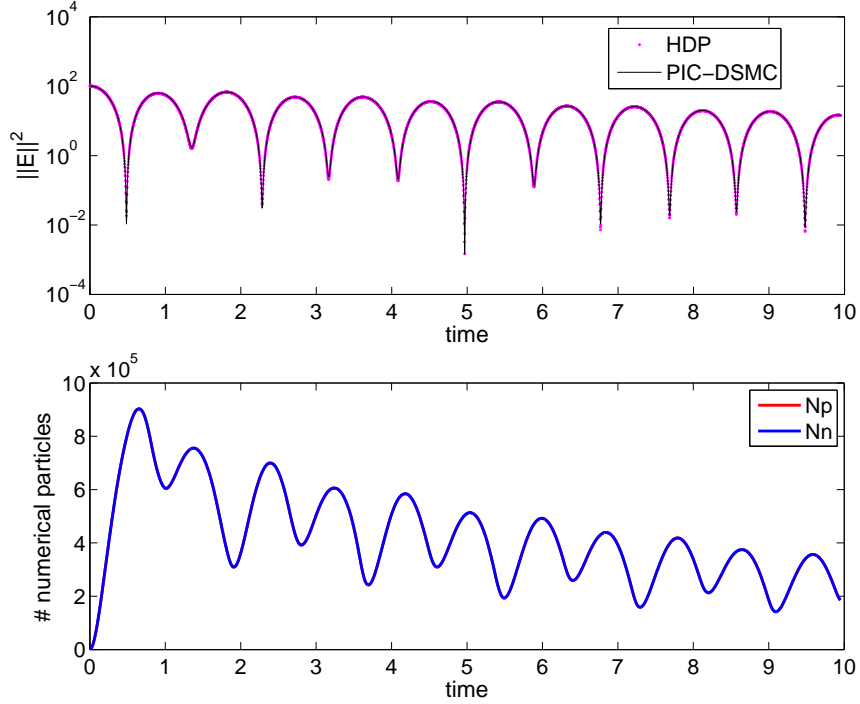


Figure 5: The nonlinear Landau damping problem of the VP-BGK system with a strong collision rate  $\mu = 10$ . Top: the comparison of the decay of energy between the HDP method (red dots) and PIC-DSMC method (black line). Bottom: the growth of the number of positive and negative deviational particles. In the bottom figure, the two lines overlap each other.

this, we consider the Landau damping problem with different  $\alpha$ 's. For each  $\alpha$ , several different effective numbers are used in computing the error (30).  $\Delta x$  and  $\Delta t$  are fixed in this test.

In Figure 12 we compare the error against the cpu time in simulation. Solutions by PIC-DSMC methods (dashed lines) and HDP methods (solid lines) with  $\alpha = 0.1, 0.01$  and  $0.001$  are computed. The three dashed lines almost overlap, suggesting that the efficiency of PIC-DSMC methods is not sensitive on whether the solution is close to the equilibrium. The solid lines give much smaller errors with the same cpu costs, illustrating the high efficiency of the HDP methods. Furthermore, Figure 12 shows the HDP methods become more efficient when the solution is closer to the equilibrium. This is due to the fact that less and less deviational and hence coarse particles are needed. The HDP methods effectively accelerate the PIC-DSMC methods in the near fluid regime.

## 5 Conclusion and Future work

We have proposed a Hybrid method with Deviational Particles (HDP) to solve the inhomogeneous Vlasov-Poisson-Landau system. This method combines several numerical techniques. The Maxwellian part is solved by a grid based fluid solver. To evolve the deviation part, which is approximated by (positive and negative) deviational particles, we apply the negative particle based Monte Carlo method proposed in [31] on the collision part and a Macro-Micro projection in [3] on the advection part. The coarse particles are simulated by a traditional PIC-DSMC method. Moreover, the particle resampling techniques are further investigated and improved.

A next step is the extension of this method to mixed regimes in which the collision frequency could range over several orders of magnitude in space. There could exist both fluid regime (where  $f \approx M$ ) and kinetic regime (where  $f$  is far away from  $M$ ). Ideally one uses only a small number of

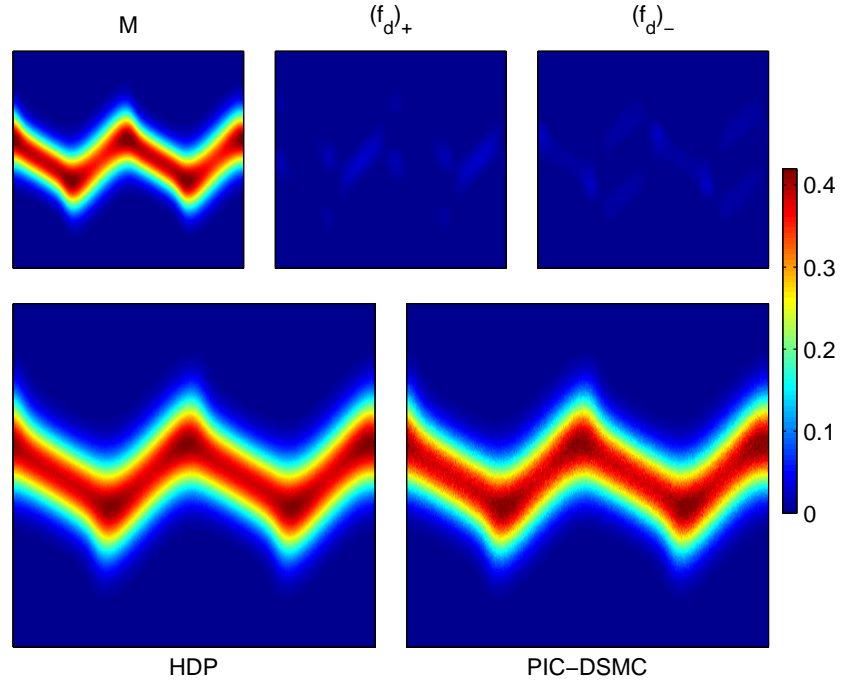


Figure 6: The snapshots of the distribution in the  $x-v_1$  phase space at time  $t = 1.25$  in the nonlinear Landau damping problem of the VP-BGK system with a strong collision frequency  $\mu = 10$ . The top three figures show the components  $M$ ,  $(f_d)_+$  and  $(f_d)_-$  in the HDP method. The bottom left figure gives the solution  $M + f_d$  obtained by HDP method. The bottom right gives the solution  $f$  obtained by PIC-DSMC method.

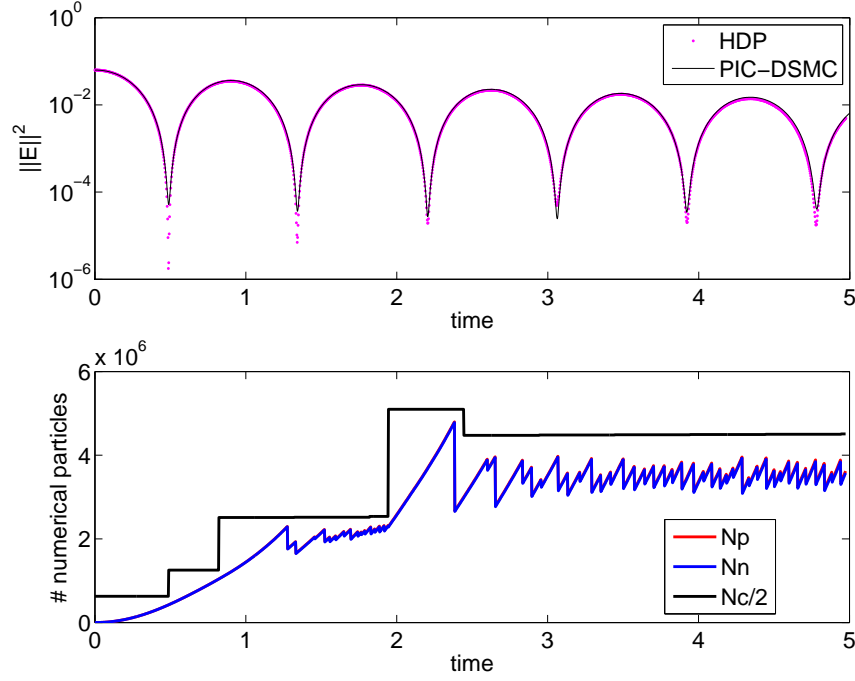


Figure 7: The linear Landau damping problem of the VPL system. Top: the comparison of the decay of energy between the HDP method (red dots) and PIC-DSMC method (black line). Bottom: the growth of the number of positive and negative deviational particles. In the bottom figure, the two lines overlap each other.

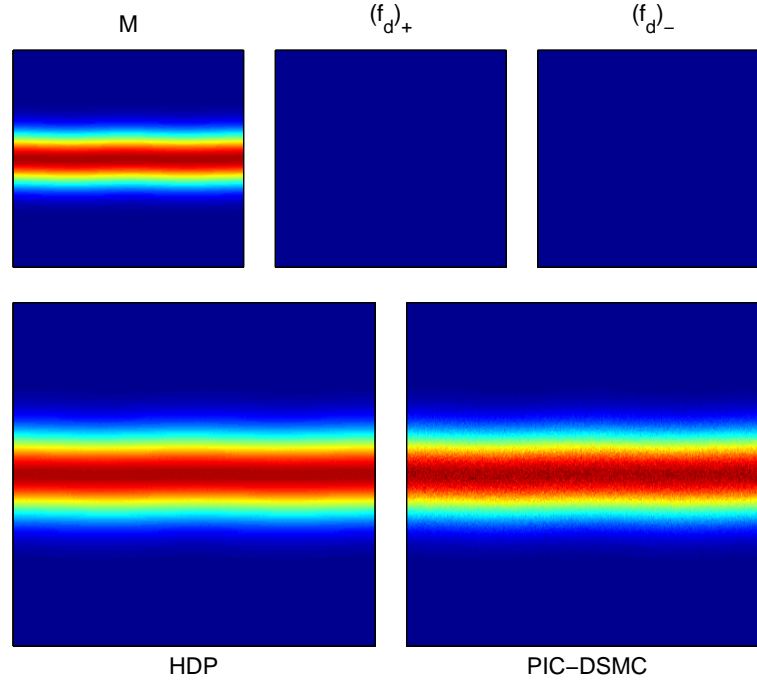


Figure 8: The snapshots of the distribution in the  $x - v_1$  phase space at time  $t = 1.25$  in the linear Landau damping problem of the VPL system. The top three figures show the components  $M$ ,  $(f_d)_+$  and  $(f_d)_-$  in the HDP method. The bottom left figure gives the solution  $M + f_d$  obtained by HDP method. The bottom right gives the solution  $f$  obtained by PIC-DSMC method.

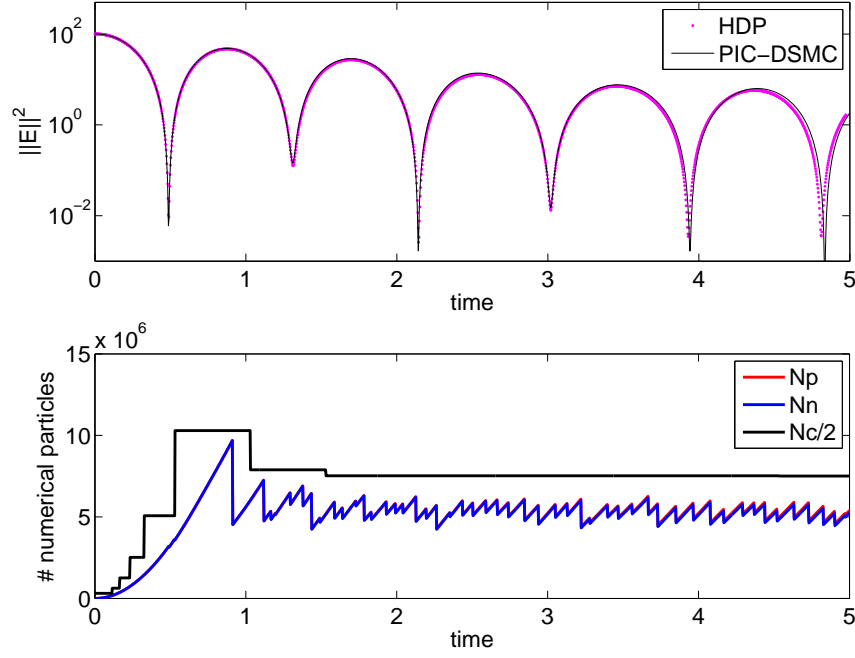


Figure 9: The nonlinear Landau damping problem of the VPL system. Top: the comparison of the decay of energy between the HDP method (red dots) and PIC-DSMC method (black line). Bottom: the growth of the number of positive and negative deviational particles. In the bottom figure, the two lines overlap each other.

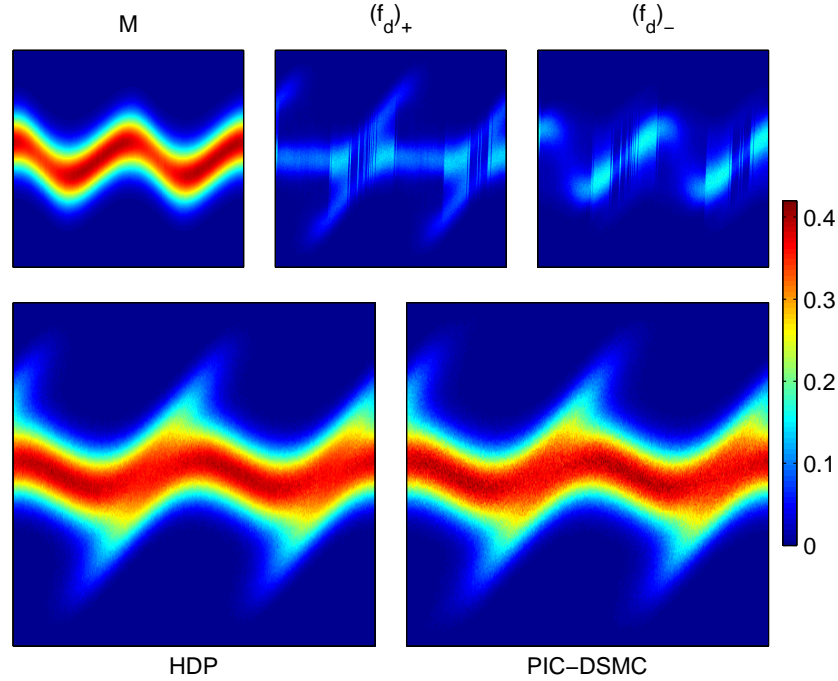


Figure 10: The snapshots of the distribution in the  $x - v_1$  phase space at time  $t = 1.25$  in the nonlinear Landau damping problem of the VPL system. The top three figures show the components  $M$ ,  $(f_d)_+$  and  $(f_d)_-$  in the HDP method. The bottom left figure gives the solution  $M + f_d$  obtained by HDP method. The bottom right gives the solution  $f$  obtained by PIC-DSMC method.

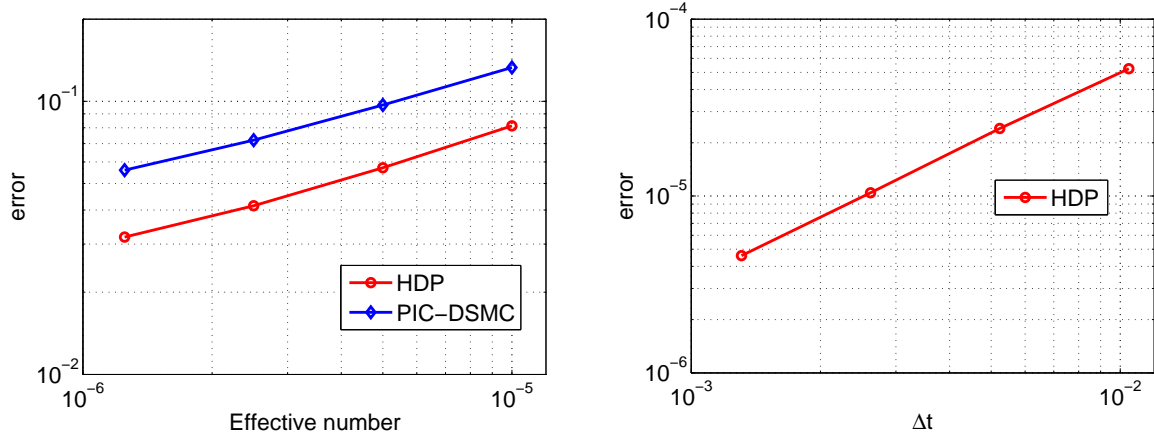


Figure 11: Convergence test of the HDP method on the VPL system. Left: the convergence in effective number in nonlinear Landau damping at time  $t = 0.5$  for the HDP method (red circles) and PIC-DSMC method (blue diamonds). Right: the convergence in  $\Delta t$  in linear Landau damping at time  $t = 0.5$  for the HDP method (red circles).

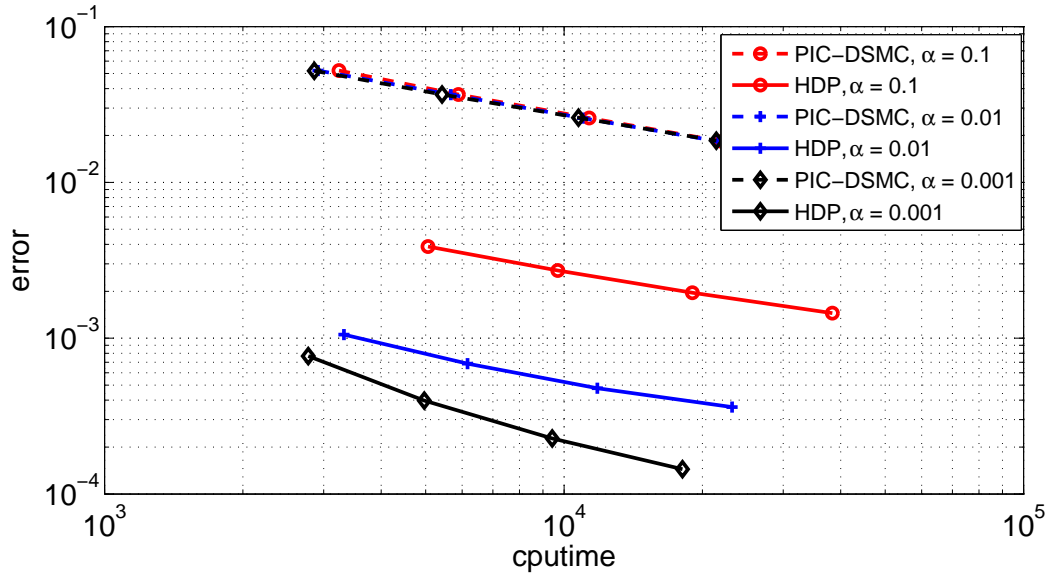


Figure 12: The efficiency test of the HDP method on the VPL system. The initial data (26) is used with different  $\alpha$ . The dash lines represent the PIC-DSMC methods while the solid lines correspond to the HDP methods.



deviational particles near the fluid regime. However in current HDP method the coarse particles have the same weights in all spatial cells. The global effective number for the coarse particles is determined by the kinetic regime where a lot of coarse particles are needed. Hence in the fluid regime, in which the same effective number is used, one also needs to evolve many coarse particles. An efficient method to transport particles with spatial variant effective numbers is to be studied in the future work.

Furthermore, as the coarse solution  $f_c$  in the HDP method for VPL system is solved by a traditional PIC-DSMC method,  $f_c$  itself is a good approximation of the solution if it is far from Maxwellian. Hence in the kinetic regime, one can discard the deviational solution ( $M + f_d$ ), and simply use  $f_c$  as the fine solution. At the interface of the two regimes, a resampling process can convert between the deviational particles and the coarse particles. This will be further investigated.

## 6 Acknowledgement

The author would like to gratefully and sincerely thank Professor Russel Caflisch in University of California, Los Angeles for his support, guidance and fruitful discussion during this work.

## References

- [1] L.L. Baker and N.G. Hadjiconstantinou. Variance reduction for Monte Carlo solutions of the Boltzmann equation. *Physics of Fluids*, 17:1–4, 2005.
- [2] L.L. Baker and N.G. Hadjiconstantinou. Variance-reduced particle methods for solving the Boltzmann equation. *J. Comput. Theor. Nanosci.*, 5:165–174, 2008.
- [3] Mounir Bennoune, Mohammed Lemou, and Luc Mieussens. Uniformly stable numerical schemes for the Boltzmann equation preserving the compressible navierstokes asymptotics. *Journal of Computational Physics*, 227(8):3781 – 3803, 2008.
- [4] G.A. Bird. *Molecular Gas Dynamics and the Direct Simulation of Gas Flows*. Oxford University Press, London, 1998.
- [5] C. K. Birdsall and A. B. Langdon. *Plasma Physics via Computer Simulation*. Institute of Physics Publishing, Bristol, UK, 1991.
- [6] A.V. Bobylev and K. Nanbu. Theory of collision algorithms for gases and plasmas based on the Boltzmann equation and the Landau-Fokker-Planck equation. *Phys. Rev. E*, 61:4576–4586, 2000.
- [7] A.V. Bobylev and I.F. Potapenko. Monte Carlo methods and their analysis for Coulomb collisions in multicomponent plasmas. *Journal of Computational Physics*, 246:123 – 144, 2013.
- [8] J. A. Byers. Noise suppression techniques in macroparticle models of collisionless plasmas. In J. P. Boris and R. A. Shanny, editors, *Proceedings of the fourth conference on numerical simulation of plasmas*, pages 496 – 510. US Govt. Printing Office, Washington, D.C., 1970.
- [9] R.E. Caflisch, C. Wang, G. Dimarco, B. Cohen, and A. Dimits. A hybrid method for accelerated simulation of Coulomb collisions in a plasma. *Multiscale Modeling and Simulation*, 7(2):865–887, 2008.
- [10] C. Cercignani. *The Boltzmann equation and its applications*. Springer-Verlag, 1988.

- [11] F. Coron and B. Perthame. Numerical passage from kinetic to fluid equations. *SIAM Journal on Numerical Analysis*, 28(1):pp. 26–42, 1991.
- [12] Anais Crestetto, Nicolas Crouseilles, and Mohammed Lemou. Kinetic/fluid micro-macro numerical schemes for Vlasov-Poisson-BGK equation using particles. *Kinetic and Related Models*, 5(4):787–816, 2012.
- [13] J. M. Dawson. Particle simulation of plasmas. *Rev. Mod. Phys.*, 55:403–447, 1983.
- [14] Richard E. Denton and M. Kotschenreuther.  $\delta f$  Algorithm. *Journal of Computational Physics*, 119(2):283 – 294, 1995.
- [15] G. Dimarco, Q. Li, L. Pareschi, and B. Yan. Numerical methods for plasma physics in collisional regimes. *Journal of Plasma Physics*, 81, 1 2015.
- [16] A.M. Dimits and W.W. Lee. Partially linearized algorithms in gyrokinetic particle simulation. *Journal of Computational Physics*, 107(2):309 – 323, 1993.
- [17] R. W. Hockney and J. W. Eastwood. *Computer Simulation Using Particles*. McGraw-Hill, 1981.
- [18] T.M. Homolle and N.G. Hadjiconstantinou. Low-variance deviational simulation Monte Carlo. *Physics of Fluids*, 19(4):–, 2007.
- [19] T.M. Homolle and N.G. Hadjiconstantinou. A low-variance deviational simulation Monte Carlo for the Boltzmann equation. *J. Comput. Phys.*, 226(2):2341–2358, October 2007.
- [20] L.D. Landau. Die kinetische gleichung für den fall Coulombscher wechselwirkung. *Phys.Z. Sowjet*, 154, 1936.
- [21] W.W. Lee. Gyrokinetic particle simulation model. *Journal of Computational Physics*, 72(1):243 – 269, 1987.
- [22] K. Nanbu. Theory of cumulative small-angle collisions in plasmas. *Phys. Rev. E*, 55:4642–4652, 1997.
- [23] J-P.M. Peraud, C.D. Landon, and N.G. Hadjiconstantinou. Monte carlo methods for solving the boltzmann transport equation. *Annual Review of Heat Transfer*, 17:205–265, 2014.
- [24] B. Perthame. *Kinetic formulation of conservation laws*. Oxford Lecture Series in Math. and its Appl. Oxford University Press, 2003.
- [25] L.F. Ricketson, M.S. Rosin, R.E. Caflisch, and A.M. Dimits. An entropy based thermalization scheme for hybrid simulations of Coulomb collisions. *J. Comput. Phys.*, 273:77C99, 2014.
- [26] M. El Safadi. Smoothness of weak solutions of the spatially homogeneous Landau equation. *Analysis and Applications*, 05(01):29–49, 2007.
- [27] Eric Sonnendrcker, Abigail Wachter, Roman Hatzky, and Ralf Kleiber. A split control variate scheme for PIC simulations with collisions. *Journal of Computational Physics*, 295:402 – 419, 2015.
- [28] T. Takizuka and H. Abe. A binary collision model for plasma simulation with a particle code. *J. Comp. Phys.*, 25:205–219, 1977.

- [29] Cédric Villani. A review of mathematical topics in collisional kinetic theory. In S. Friedlander and D. Serre, editors, *Handbook of Mathematical Fluid Dynamics*, volume 1, pages 71 – 305. North-Holland, 2002.
- [30] W. Wagner. Deviational particle Monte Carlo for the Boltzmann equation. *Monte Carlo Methods and Applications.*, 14(3):191–268, 2008.
- [31] Bokai Yan and Russel E. Caflisch. A Monte Carlo method with negative particles for Coulomb collisions. *J. Comput. Phys.*, 298:711–740, 2015.

## A Compute the source term in (22c)

In this appendix we show how to efficiently sample deviational particles from the source term in the right hand side of (22c).

Suppose we have the moments of the Maxwellian at time  $t^n$  and the electric field  $\mathbf{E}$  has been solved from the Poisson equation (1b). Then the term  $\mathcal{T}M$  is just the Maxwellian  $M$  multiplied by a cubic polynomial in  $\mathbf{v}$ ,

$$\begin{aligned}\frac{\mathcal{T}M}{M} &= \frac{1}{M}(\mathbf{v} \cdot \nabla_{\mathbf{x}} M - \mathbf{E} \cdot \nabla_{\mathbf{v}} M) \\ &= \mathbf{v}^T \left( \frac{\nabla_{\mathbf{x}} \rho}{\rho} + \frac{\nabla_{\mathbf{x}} T}{2T} \left( \frac{|\mathbf{v} - \mathbf{u}|^2}{T} - d \right) + \nabla_{\mathbf{x}}^T \mathbf{u} \frac{\mathbf{v} - \mathbf{u}}{T} \right) + \mathbf{E}^T \cdot \frac{\mathbf{v} - \mathbf{u}}{T},\end{aligned}$$

with the matrix

$$\nabla^T \mathbf{u} = [\nabla_{\mathbf{x}} u_1, \nabla_{\mathbf{x}} u_2, \nabla_{\mathbf{x}} u_3].$$

Here  $\rho$ ,  $\mathbf{u}$  and  $T$  are the density, macroscopic velocity and temperature of the Maxwellian part.

To compute  $\Pi_M \mathcal{T}M$ , from the definition (20), one needs to evaluate the the following coefficients

$$\begin{aligned}\langle \mathcal{T}M \rangle &= \nabla_{\mathbf{x}} \cdot (\rho \mathbf{u}), \\ \langle (\mathbf{v} - \mathbf{u}) \mathcal{T}M \rangle &= \rho(\mathbf{u} \cdot \nabla_{\mathbf{x}}) \mathbf{u} + \nabla_{\mathbf{x}}(\rho T) + \rho \mathbf{E}, \\ \left\langle \left( \frac{|\mathbf{v} - \mathbf{u}|^2}{T} - d \right) \mathcal{T}M \right\rangle &= \frac{\rho d}{T} \mathbf{u} \cdot \nabla_{\mathbf{x}} T + 2\rho \nabla_{\mathbf{x}} \cdot \mathbf{u}.\end{aligned}$$

Similarly, for the term  $\Pi_M(\mathcal{T}f_d)$ , one needs to evaluate the following moments for  $\phi = 1, \mathbf{v}, |\mathbf{v}|^2$ ,

$$\langle \phi \mathcal{T}f_d \rangle = \nabla_x \cdot \langle \mathbf{v} f_d \phi \rangle,$$

where we have used  $\langle \phi \mathbf{E} \cdot \nabla_{\mathbf{v}} f_d \rangle = 0$ . Note that this has been computed in (19) to update  $M$  via (22b) and can be reused.

Since the projection  $\Pi_M \psi$  is just the Maxwellian  $M$  multiplied by a quadratic function in  $\mathbf{v}$ , the whole source term in (22c) is in the form of  $P_3(\mathbf{v})M(\mathbf{v})$ , with  $P_3(\mathbf{v})$  some cubic function in  $\mathbf{v}$ . One can easily find another Maxwellian with slightly larger temperature as an upper bound. Then deviational particles can be sampled efficiently by a regular rejection-acceptance method.

## B Acceleration of Particle Resampling

In this section we describe an approximation of the resampling formula (23) which greatly accelerates the resampling process. We also test on the efficiency numerically.

## B.1 Acceleration Technique

We use the Fourier basis  $\phi_{\mathbf{k}}(\mathbf{v}) = \exp(-i\mathbf{k} \cdot \mathbf{v})$  in the expansion (23). Note that the evaluation of each Fourier coefficient

$$\hat{f}_{\mathbf{k}} = N_{\text{eff}} \left( \sum_{\mathbf{v}_p} \phi_{\mathbf{k}}(\mathbf{v}_p) - \sum_{\mathbf{v}_n} \phi_{\mathbf{k}}(\mathbf{v}_n) \right)$$

needs to compute  $\mathcal{O}(N_d)$  exponents. This leads to the expensive cost of  $\mathcal{O}(N_d K^3)$  to evaluate all Fourier coefficients in the 3-dimensional velocity space. In the following we give a much cheaper approximation of  $\hat{f}_{\mathbf{k}}$ .

After shifting and rescaling, we can assume all the deviational particles are located in the cube  $[0, 2\pi]^3$  in the velocity space. Place a uniform grid on  $[0, 2\pi]^3$  with  $\tilde{K} (\geq K)$  points in each direction. For each positive deviational particle  $\mathbf{v}_p$ , write it as

$$\mathbf{v}_p = \mathbf{v}_{\xi(p)} + \delta_p,$$

where  $\mathbf{v}_{\xi(p)}$  is the nearest grid point of  $\mathbf{v}_p$ . Let  $I_{\mathbf{j}} = \{p : \xi(p) = \mathbf{j}\}$ , which is the set of all particles near the  $\mathbf{j}$ th grid. Define the following functions on the grids

$$\begin{aligned} f^{(0)}(\mathbf{v}_{\mathbf{j}}) &= N_{\text{eff}} \sum_{p \in I_{\mathbf{j}}} 1, \\ f^{(1)}(\mathbf{v}_{\mathbf{j}}) &= N_{\text{eff}} \sum_{p \in I_{\mathbf{j}}} \delta_p, \\ f^{(2)}(\mathbf{v}_{\mathbf{j}}) &= N_{\text{eff}} \sum_{p \in I_{\mathbf{j}}} \delta_p \delta_p^T. \end{aligned}$$

Note that the vector  $f^{(1)}$  has 3 components and the matrix  $f^{(2)}$  has  $3^2$  components. Then the first term in the coefficient of (23) is

$$\begin{aligned} & N_{\text{eff}} \sum_{\mathbf{v}_p} \exp(-i\mathbf{k} \cdot \mathbf{v}_p) \\ &= N_{\text{eff}} \sum_{\mathbf{v}_p} \exp(-i\mathbf{k} \cdot \mathbf{v}_{\xi(p)}) \exp(-i\mathbf{k} \cdot \delta_p) \\ &= N_{\text{eff}} \sum_{\mathbf{v}_p} \exp(-i\mathbf{k} \cdot \mathbf{v}_{\xi(p)}) \left( 1 - i\mathbf{k} \cdot \delta_p - \frac{1}{2}(\mathbf{k} \cdot \delta_p)^2 \right) + \mathcal{O}(|\mathbf{k}| \Delta v / 2)^3 \\ &= N_{\text{eff}} \sum_{\mathbf{j}} \exp(-i\mathbf{k} \cdot \mathbf{v}_{\mathbf{j}}) \sum_{p \in I_{\mathbf{j}}} \left( 1 - i\mathbf{k} \cdot \delta_p - \frac{1}{2}(\mathbf{k} \cdot \delta_p)^2 \right) + \mathcal{O}(|\mathbf{k}| \Delta v / 2)^3 \\ &= \sum_{\mathbf{j}} \exp(-i\mathbf{k} \cdot \mathbf{v}_{\mathbf{j}}) \left( f^{(0)}(\mathbf{v}_{\mathbf{j}}) - i\mathbf{k} \cdot f^{(1)}(\mathbf{v}_{\mathbf{j}}) - \frac{1}{2} \mathbf{k}^T f^{(2)}(\mathbf{v}_{\mathbf{j}}) \mathbf{k} \right) + \mathcal{O}(|\mathbf{k}| \Delta v / 2)^3 \\ &= \hat{f}_{\mathbf{k}}^{(0)} - i\mathbf{k} \hat{f}_{\mathbf{k}}^{(1)} - \frac{1}{2} \mathbf{k}^T \hat{f}_{\mathbf{k}}^{(2)} \mathbf{k} + \mathcal{O}(|\mathbf{k}| \Delta v / 2)^3. \end{aligned} \tag{31}$$

Therefore one can first recover the functions  $f^{(0)}$ ,  $f^{(1)}$  and  $f^{(2)}$  on the grids, which takes a number of  $\mathcal{O}(N_d)$  computations of polynomials. Then apply a Fast Fourier Transform to each function and multiply by  $\mathbf{k}$  and  $\mathbf{k}\mathbf{k}^T$ , which costs  $\mathcal{O}(\tilde{K}^3 \log \tilde{K})$ . Only a cost of  $\mathcal{O}(N_d + \tilde{K}^3 \log \tilde{K})$  is needed to get an approximation. In practice we take  $\tilde{K} = 2K$ . With  $K = 30$ , the omitted high order term is  $\mathcal{O}((\tilde{K})^{-3}) \approx 10^{-6}$ , when evaluating the low modes.

The second term in the coefficient of (23) can be approximated in a similar way on the negative deviational particles. Once the explicit formula (23) is recovered, one can use a similar approximation to evaluate the function  $f(\mathbf{v})$  for each sampled particle with velocity  $\mathbf{v}$ .

## B.2 Numerical Test

We perform a numerical test on the resampling of deviational particles in a single cell. The positive and negative deviational particles are sampled from two Maxwellians in 3D velocity space with

$$\rho_p = 1, \mathbf{u}_p = 0, T_p = 2, \rho_n = 1, \mathbf{u}_n = 0, T_n = 1.$$

$10^7$  particles are sampled from each distribution.  $K = 30$  modes are used. Figure 13 shows the distribution  $|\mathbf{v}|^2 f_d(\mathbf{v})$  in the energy space and the  $2k$ -th moments  $\int (v_1)^{2k} f_d(\mathbf{v}) d\mathbf{v}$  before and after resampling with the approximation (31). A good agreement is observed. This single resampling process takes about 20 seconds by a serial C++ code on a single core 2.0 GHz CPU processor. As for the resampling with the full expansion (23), we find it cannot be finished in 24 hours on the same processor if  $10^7$  deviational particles are used. For comparison, with  $10^4$  deviational particles, the resampling with the full expansion takes about 5 minutes.

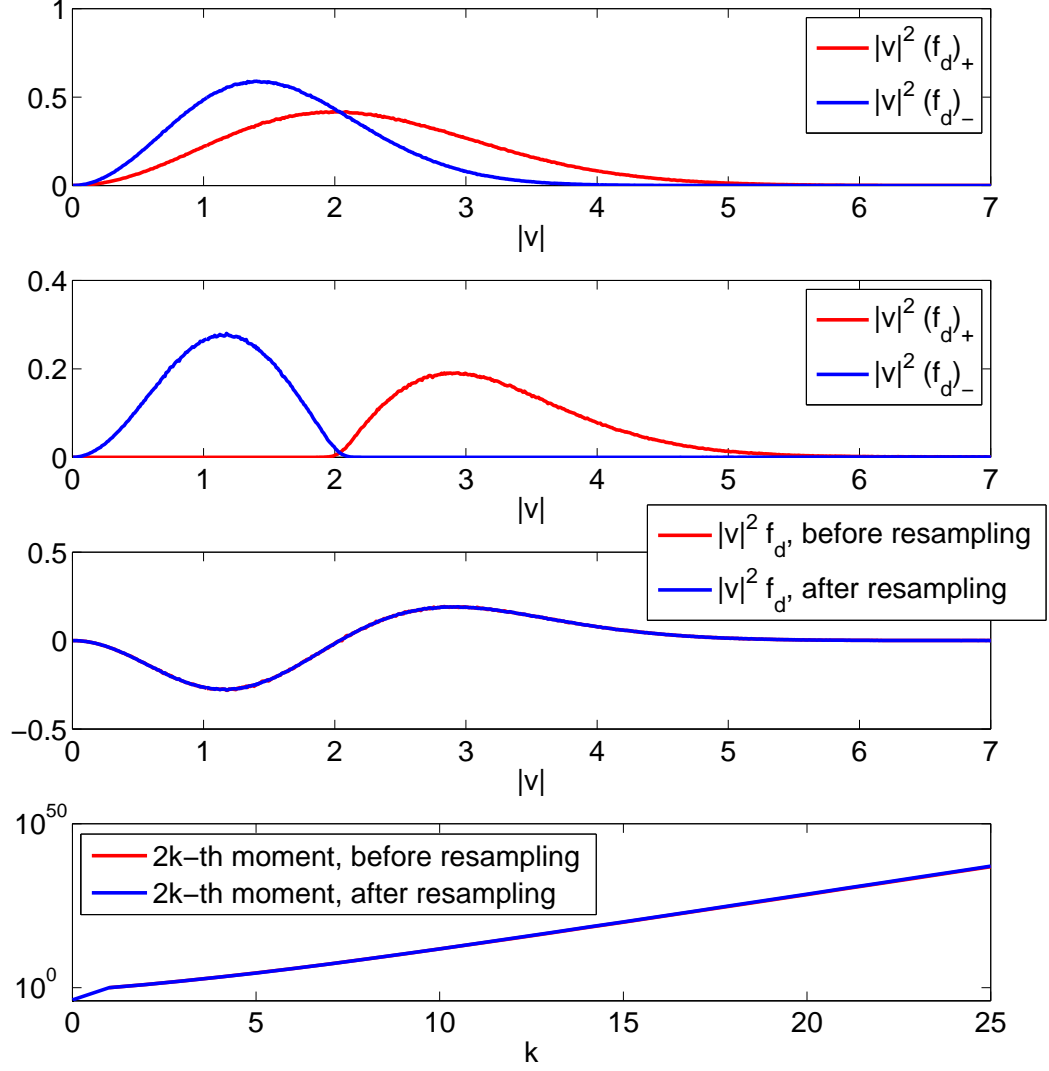


Figure 13: The resampling with the approximation (31). From top to bottom: the distributions  $|v|^2(f_d)_+$  and  $|v|^2(f_d)_-$  before resampling; the distributions  $|v|^2(f_d)_+$  and  $|v|^2(f_d)_-$  after resampling; the distribution  $|v|^2 f_d$  before and after resampling; the moments  $\int (v_1)^{2k} f_d(v) dv$  before and after resampling.